Energy and task quality optimisation in WSN systems

Niall Creech[a], Natalia Criado ~~Pacheco~~[b] and Simon Miles[c]

[a]*Department of Informatics, King's College London, Bush House, Strand Campus, 30, Aldwych, London WC2B 4BG*
[b]*Department of Informatics, King's College London, Bush House, Strand Campus, 30, Aldwych, London WC2B 4BG*
[c]*Department of Informatics, King's College London, Bush House, Strand Campus, 30, Aldwych, London WC2B 4BG*

ARTICLE INFO

Keywords:
multi-agent system
multi-agent reinforcement learning
wireless sensor network

ABSTRACT

Wireless sensor networks (WSN) are applicable across a wide range of industries from vehicle-to-vehicle communications to large-scale environmental monitoring. They need to manage energy usage to maintain availbility, distribute tasks effectively, and handle distruption to network changes and agent loss. Decentralised algorithms are commonly used to meet these challenges, with hierarchical cluster formation or reinforcement learning techniques. There are challenges however in getting these algorithms to perform well in large distributed systems where there are multiple objectives, dynamic agents, or connectivity change. In this work we propose the novel agent networks with hierarchical task allocation optimisation (AN-HTAO) algorithm to optimise WSN systems based on the multiple objectives of maximising energy availability, distribution, and task quality, while maintaining task coverage in a dynamic network. This integrates and extends upon the previously defined agent task allocation with risk-impact awareness (ATA-RIA) and multi-group resource allocation optimisation (MG-RAO) algorithms, adding hierarchical task allocation with multi-objective composite task value calculation. We evaluate the algorithms' performance in environmental monitoring-based simulated systems where there are a number of measurement tasks to be completed within the system. The AN-HTAO algorithm showed a 22% system utility improvement in the simple 10 node system, and 14% in a more complex 25 node system over 500 episodes. Energy availability was increased by 6% and 29% respectively. Evaluation of changing algorithm parameters to balance between energy availability, distribution, and task quality showed that these individual components could be prioritised in different ratios depending on the requirements of the optimisation required in the system.

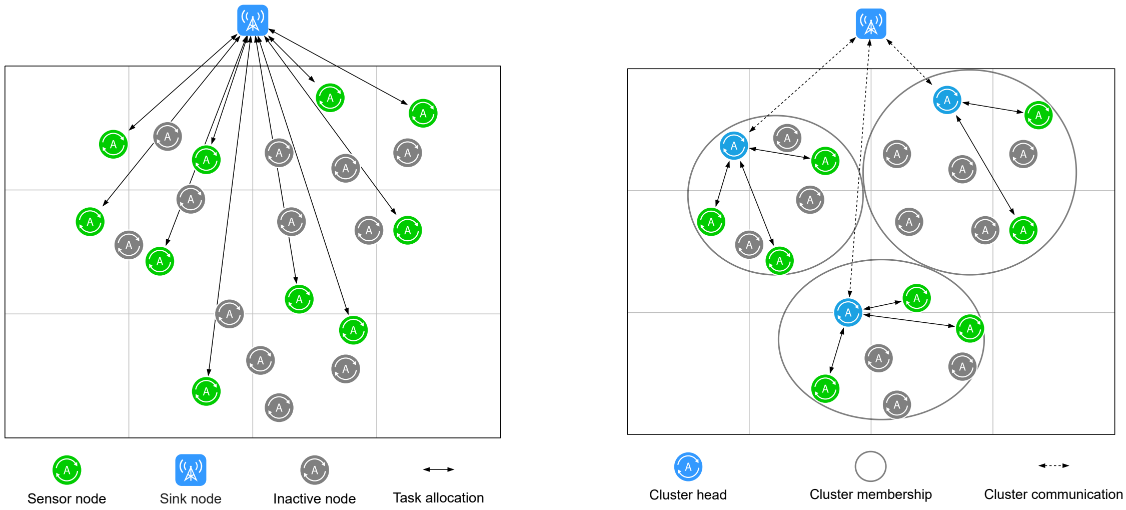
1. Introduction

Wireless sensor networks (WSNs) have many applications and research studies in areas such as monitoring the environment such as oceanographic measurement (Mahdy, 2008; Albaladejo et al., 2010a; Xu et al., 2014), radioactive contamination (Gomez et al., 2015), water quality (Fang et al., 2010), flood levels (Castillo-effen et al., 2004), volcanic activity (Werner-Allen et al., 2006), agricultural soil (Goel and Bindal, 2018), as well as in military uses (Đurišić et al., 2012). More recently, the availability and lower cost of low-power wireless transmitters (Min et al., 2001), solar-harvesting components (Prauzek et al., 2018), and micro-electro-mechanical systems (Warneke and Pister, 2002) has allowed large deployments sizes and scope of use, expanding their real-world use and opening up new areas for practical research (Corke et al., 2010; Kandris et al., 2020).

WSNs can be structured with centralised, or decentralised control. With centralisation, the controller node has system-wide knowledge, and so allocates measurement tasks to other nodes, orchestrates their communications, and handles recovery (see Figure 1a). This approach inherently does not scale to large networks due to congestion and resource exhaustion on the central component. In harsh environments, this centralisation of control is not robust to damage or node loss. Although there can be learning added to these systems to help optimise in complex systems, non-distributed learning algorithms suffer from the same limitations as non-learning WSN systems (Predd et al., 2006). For these reasons, we focus on decentralised, autonomous methods for WSN construction.

With decentralised WSNs, nodes have limited knowledge of the system. Each node acts autonomously to some degree to orchestrate the functionality mentioned above. They are often organised into groups to decrease the cost of coordination without full centralisation. The basis of many of these decentralised techniques is hierarchical, typically

*Corresponding author
niall.creech@kcl.ac.uk (N. Creech); natalia.criado_pacheco@kcl.ac.uk (N.C. Pacheco); simon.miles@kcl.ac.uk (S. Miles)
ORCID(s): 0000-0002-9573-0991 (N. Creech)



(a) A centralised WSN configuration with direct communication (b) A decentralised WSN configuration based on clustering

Figure 1: Common WSN configurations for task coverage of a grid

some form of clustering technique. Nodes are formed into sub-groups with designated leaders that orchestrate the behaviours of each group of nodes and communicates with the central controller (see Figure 1b) . This ability for nodes to work autonomously increases resilience, however, it does introduce new challenges. In working with local information only, the system can fail to optimise well or get trapped in local optima. While broad, system-wide coordination of node behaviours are difficult with communication limited to small sets of neighbours (Carlos-Mancilla et al., 2016a).

Reinforcement learning has seen applications in these decentralised systems, often focussed on using adaptive routing to optimising energy efficiency and system lifetime (Mihaylov et al., 2012; Kulkarni et al., 2011), or to ensure sensor coverage is maintained with the loss of nodes over time (Sharma and Chauhan, 2020). These solutions look at single, or complimentary objective optimisation, and do not take account of the conflicting multiple objectives in WSN systems. System goals may involve energy optimisation, coverage, measurement quality, and how nodes allocate resources to meet these goals. The work we present tackles the optimisation of multiple, competing objectives in a WSN system. While also adapting network routes dynamically to not only optimise for energy, but also in response to task quality and network impacts.

The solution we present here is based on the following algorithms previously developed by the authors. We use the ATA-RIA algorithm to optimise the task of measurements and coverage, minimise the energy consumption of the network, while adapting to the dynamic nature of WSNs (Creech et al., 2021a). Through the MG-RAO algorithm we enable sensors that are taking measurements to optimise the allocation of their resources to meet the overall system goal (Creech et al., 2021b). By combining and evaluating these algorithms in a simulated WSN deployed in a realistic environment, we show that the overall solution can be successfully utilised to balance the systems' multiple objectives of minimising energy consumption, maximising system lifetime, as well as optimising the quality of the measurement tasks while still maintaining geographical coverage.

In Section 2 we look at related research in this area, allowing us to concretely define the problem in Section 3. Section 4 sets out our solution, followed by definition of the simulated environment and evaluation of the solution in Section 5. We close with the summary of our conclusions and future work in Section 6.

2. Related work

Wireless sensor networks (WSNs) are collections of independent battery-powered sensor nodes connected through wireless transmission, used to monitor a geographical area (Akyildiz et al., 2002; Yick et al., 2008). In many circumstances the areas under study are difficult to access, such as remote locations, ocean-based studies, or contaminated areas. *Environmental wireless sensor networks (E-WSNs)* in particular are characterised by intermittent connectivity between nodes, and harsh environmental conditions that lead to degradation of the network, so that the network must be adaptable in response to these changes (Oliveira and Rodrigues, 2011). In these circumstances batteries are non-replaceable, and dispersal is ad-hoc, making a high level of autonomous node communication, coordination, and power consumption management, essential. WSNs can consist of static sensor deployments, or as mobile agents, adding to the need for adaptation in the network (Ramasamy, 2017; Munir et al., 2007). Sensors are usually small and inexpensive, capable of utilising limited compute and storage resources. They have one or multiple sensing devices to take measurements from the environment, ranging through chemical, optical, thermal, biological, and radioactivity detection. This collected data can then be transmitted to a base station and retransmitted to be stored remotely and analysed.

In realistic scenarios, there are five key elements that present challenges in deploying and operating a WSN system.

1. *Energy consumption.* Each node in a WSN network has limited power available to function supplied by a battery. Depending on the environmental conditions, there may be some form of energy-capture component built into each node, such as solar-harvesting (Prauzek et al., 2018). However, even with this additional energy replenishment, energy is limited, and will eventually be exhausted. Batteries cannot be manually replaced in remote or inhospitable locations, that are often the focus of WSN deployments, so minimising energy consumption is essential (Anastasi et al., 2009). This can be done through the use of low-power components, (Sonavane et al., 2008; Wu et al., 2017), as well as applying energy-aware routing protocols amongst nodes (Liu et al., 2009).
2. *System lifetime.* Whether due to the impact of environmental degradation, power exhaustion, or connectivity loss, nodes in a network have a limited useful lifespan (Mak and Seah, 2009). As nodes are lost, the system itself becomes degraded, eventually unable to achieve its task to a high enough quality. At this point, the systems' lifetime has been reached and it is no longer useful. To extend this lifetime as far as possible we try to reduce the wear on nodes, principally, by ensuring that energy consumption is distributed through out the system evenly (Babayo et al., 2017; Engmann et al., 2018).
3. *Quality of measurement data.* A node taking a reading may have faulty sensor, leading to variations and lack of reliability. Sensors may get more accurate readings using more energy or longer time scales, for example, as the sampling time of a temperature or radiation sensor is increased, the more accurate the reading (Bhandari et al., 2017). Therefore nodes must trade-off the quality of its data acquisition, with the restrictive energy available to it across its lifetime (Dias et al., 2016).
4. *Coverage of sensors.* In the majority of environmental situations, sensors are distributed in an ad-hoc manner, meaning their distribution is initially unknown amongst the nodes. Sensors may also have occlusion problems due to the topography of the environment or objects blocking connectivity or measurement (Qian and Qi, 2008). Therefore, to initialise the system, the deployed nodes must find which other nodes to communicate to that will allow all demand points to be covered. They must also be resilient to temporary or permanent outages on the network that means a new communication network must be discovered.
5. *Routing failure* Wireless sensor networks add substantial additional risks to reliability over standard networking. Nodes are at risk of running out of power or of component failure, often exacerbated by harsh conditions. Environmental effects or obstacles may physical impact transmission or reception of signals. Loss of communication to a node is especially impactful as there are often multi-hop routes involved, which multiplies the risks (Paradis and Han, 2007).

With the unknown nature of optimal network formation, task allocation, and resource allocation at initialisation, as well as its variability throughout the lifetime of the system, the use of learning strategies has had active research. Machine learning and reinforcement learning has been applied to many of the key requirements, routing in ad-hoc networks (Nurmi), allocating tasks such as sensor measurement (Khan and Rinner, 2014), balancing the energy consumption (Mihaylov et al., 2010; Praveen Kumar et al., 2019), and to the balancing of these multiple system objectives overall (Sengupta et al., 2013; Iqbal et al., 2015).

By setting the system objectives as the minimisation of energy consumption, and the maximisation of system lifetime, measurement quality, and sensor coverage, we can define the problem as a multi-objective learning problem in a distributed multi-agent system. In the next section we formalise the terminology, definitions, and the problem we look to address.

3. Balancing energy and measurement in a WSN system

In this section we will formally set out the problem we look to solve. Firstly, we define our WSN system in terms of agents, available resources, and tasks to be completed. Then we go on to look at the roles different agents play in

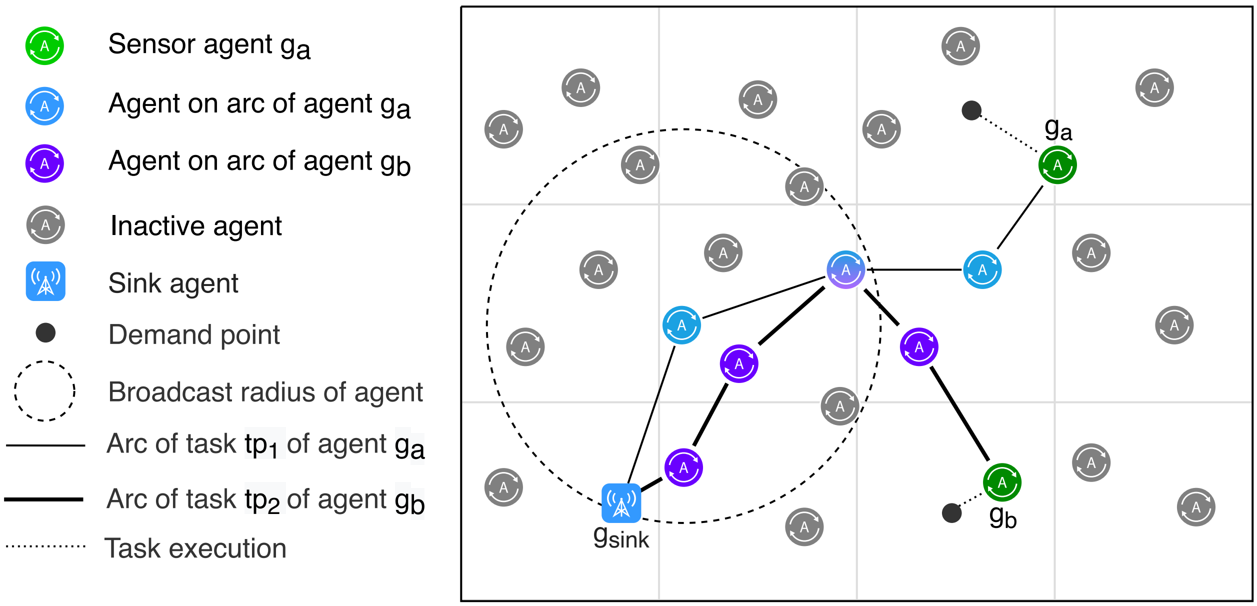


Figure 2: WSN components and terminology

completing a measurement task, and how energy is consumed in doing so. We next detail task coverage, and how this relates to the resilience of the network to impacts and failures. Finally, we define to what quality a measurement is completed, and how this, energy availability, and energy distribution in the system, effects the quality of overall composite task completion.

3.1. Wireless sensor network components and terminology

A WSN system is comprised of a set of *nodes* or *agents*. Each agent is equipped with a microcontroller for computation, a battery for power storage, a wireless transceiver for transmitting and receiving messages from other agents, and one or more *sensors* for sensing and measuring some property of the environment such as temperature or radiation levels (Ahmed et al., 2012). The agents may be deployed to precise locations or through more random distribution. The network structure formed may be flat or hierarchical, with or without clustering, dependent on the choice of protocol used (Carlos-Mancilla et al., 2016b). Requests for measurement from outside the WSN are received from an capable agent, a *sink node*, which may then relay these requests through other agents until a final measurement is made by a *sensing node*. This sequence of agents comprising the measurement task execution are defined as the *task arc*. These concepts are illustrated in Figure 2 and will be detailed in the following sections.

3.2. Tasks and resources

In our assumed system there is a geographical area to monitor defined by a two dimensional grid of real numbers on which a set of agents G are distributed randomly. The *deployment configuration* is a mapping from agents to their respective locations on this grid, $conf : G \to (\mathbb{R} \times \mathbb{R})$. Agents perform tasks, which are typed. These can be either *atomic tasks*, for individual measurements, or *composite tasks*, composed of sets of atomic tasks. Each atomic task targets a measurement at a location on this grid, its *demand point* $dp(at)$.

Composite tasks are allocated from outside the system to sink nodes, throughout the systems lifetime. These composite tasks are then decomposed by agents and the corresponding atomic tasks either completed by that agent, or allocated to other agents to complete or relay to further agents. Completing atomic tasks requires *resources*, such as energy, of which an agent has a fixed amount defined by $ar : G \times \mathbb{R} \to \mathbb{R}_{>=0}$. At at time ϕ, each agent carrying out an atomic task has a certain amount of the resources it possesses r, assigned to completion of tasks of that type, given by $ra_g : AT \to 2^{R \times \mathbb{R}_{>=0}}$

We can therefore define the system as a tuple $\langle AT, CT, G, R, ar, sg, conf \rangle$, where

- G is a set of agents in the system that can complete tasks;

- $SG \subset G$ is a set of sink nodes, agents that can receive composite tasks from external to the system;
- AT is a set of atomic tasks where each task is a measurement task performed by a single agent;
- $CT \subseteq 2^{AT}$ is the set of composite tasks that occur in the system;
- R is a set of resources needed to perform atomic tasks;
- $ar : G \times \mathbb{R} \to \mathbb{R}_{>=0}$ is a mapping from each agent and each resource to the amount of that resource that the agent possesses.
- $sg : CT \to 2^{SG}$ is a mapping from each composite task type to the group of sink agents that receive and ensure the completion of tasks of that type.
- $conf : G \to (\mathbb{R} \times \mathbb{R})$: is a mapping of each agent to their location in the system.

3.3. Node roles and task arcs

To simplify discussion of task execution we distinguish agents by the role they play in a given atomic task. These roles will also define the energy used by the nodes in executing a composite task and component atomic tasks (Gupta and Sinha, 2014).

- A *sink node* of an atomic task *at* is the agent that first receives the corresponding composite task, and will broadcast the results, $sink : AT \to G$.
- A *sensing node* is the agent that executes the atomic task and so performs the sensor measurement, $sensor : AT \to G$.
- An *active node* is an agent that participates in sub-allocating, or routing, that task, but is neither a sink agent nor a sensing agent, $active : AT \to 2^G$..
- An *idle node* does not participate in the specific task, but does in other tasks in the system during a time period Φ, $idle_{\Phi} : AT \to 2^G$.
- An *sleeping node* does not participate in any of the tasks in the system during a time period Φ, $sink_{\Phi} : AT \to 2^G$.

With these roles in mind, we can now define the *task arc* as a mapping of atomic tasks to ordered sequence of agents $arc : AT \to 2^G$ that each atomic task *at* is sub-allocated to. The first agent is the sink node that has received the initial composite task, and the last agent is the sensing agent that executes the atomic task, with the sequence of agents in-between relaying the atomic task. So that, for an arc of length n, we have $arc(at) = \{sink(at), active_i(at), sensor(at)\}_{i=1}^{n-1}$.

3.4. Energy consumption and availability

The *atomic task energy consumption* is the energy used by all agents in the system in executing an atomic task *at*. This is composed of transmission energies, TE, the power used by nodes to broadcast a message to allocate a task to another node, or reply with a task result. Receiver energy, RE, is the energy used by an agent to receive a message. Both transmission and receiving in an atomic task sequence of allocations involve all members of the arc twice, apart from the sink and sensing nodes which will only transmit and receive once. Note, we make the assumption that the energy of a broadcast of receipt of a message is independent of range so that the energy use per-atomic task is

$$2TE(|arc(at)| - 1) + 2RE(2|arc(at)| - 1) \tag{1}$$

Although nodes would still use energy when in an idle power saving mode, or sleep mode, we disregard these for this formulation as we look to optimise the active node power usage only. There are other algorithms that can be used to optimise the cycling of power cell usage (Escolar et al., 2014).

The *fractional agent energy* maps the an agent to its available energy, as a fraction of the batteries' maximum capacity, $fae : G \to \mathbb{R}[0, 1]$. We can then specify the *fractional energy availability*, as the sum-average of the agent energy of all agents in a set G.

$$ea_{\phi}(G) = \frac{\sum_{\forall g \in G} fae_{\phi}(g)}{|G|} \tag{2}$$

To optimise for distribution of energy usage we minimise the variance[1] of the fractional agent energy values. As we look to optimise by maximising across multiple goals, and the values of $fae_\phi(g)$ are bounded by $[0, 1]$, we can rephrase this as maximising the distance between the variance and the maximum possible, $1/4$. So we use the *relative energy variability* as our distribution measurement,

$$rev_\phi(G) = \frac{1}{4} - \sigma^2(\{fae_\phi(g)\}_{\forall g \in G}) \tag{3}$$

3.5. Coverage and resilience of routing

Since network impacts will mean that some nodes will not be able to be allocated an atomic task, temporarily or permanently, an agent trying to complete a composite task may not be able to do so. We can therefore use the proportion of atomic tasks of a given composite task that are completed successfully as a measure of how resilient the network is, how well it has adapted to impact and loss of nodes. For each completed composite task ct, we have a corresponding set of successfully completed atomic tasks, $ct^* \subseteq ct$, so the *coverage* of ct is simply $|ct^*|/|ct|$. We can use this to measure the *system coverage* over a time period Φ, the average coverage of all composite tasks completed in during that period

$$cov_\Phi(CT, CT^*) = \sum_{\forall ct \in CT} \frac{|ct^*|/|ct|}{|CT|} \tag{4}$$

3.6. Task quality and system utility

An agent that takes a measurement will complete that task with a certain *atomic task quality*, a quality that is dependant on the resources it has dedicated to tasks of that type, and its distance to the tasks demand point. For example, a sensor is capable of taking a measurement of the radiation levels at the location of its sensor, which may be a distance away from the tasks' demand point. In addition, the longer the sample time the more energy is used, but the more accurate the reading will be. We therefore specify the atomic task quality of an atomic task *at* as a function of the distance of the task from the requested position, and the amount of energy used to make the measurement.

$$q_{g,\phi}(at) = ra_g(at) \times |dp(at) - conf(g)| \tag{5}$$

The *composite task value*, the proportional value of an atomic task *at* to the corresponding composite task *ct*, will not only be dependent on the quality of the atomic task, but also on additional system wide objectives. In a WSN system we want to maximise values for multiple objectives. Firstly, the energy available in the system, therefore minimising the overall battery power consumption. Secondly, the distribution of energy use across the nodes, which will spread component wear across nodes and so maximise the usable lifetime of the system. Lastly, the quality of atomic tasks, which not only improves the accuracy of readings, but will also invoke a penalty of poor task coverage as failed tasks have lower quality. Therefore the *composite task value* of a composite task *ct* is defined with these multiple objectives,

$$ctv(at) = \alpha \underbrace{ea_\phi(arc(at))}_{\text{energy available}} + \beta \underbrace{rev_\phi(arc(at))}_{\text{energy distribution}} + \gamma \underbrace{q_{g,\phi}(at)}_{\text{task quality}} \tag{6}$$

Where α, β, and γ are variables chosen at system initialisation to weight the influence of energy, distribution, and task quality respectively. With the composite task value calculated after its completion at a time ϕ, we can then calculate *composite task quality* as the combination of each atomic tasks quality and its value to the composite task,

$$taq_\phi(ct) = \sum_{\forall at \in ct} ctv(at) q_{g,\phi}(at) \tag{7}$$

Finally we can define each atomic task absolute value to the system as the product of the respective composite tasks' quality and the relative contribution of the atomic task to that quality,

$$atv(ct, at) = taq_\phi(ct) ctv(at) \tag{8}$$

We so define the overall *utility* of the system over a time period Φ as the sum of the composite task qualities of all the composite tasks CT completed during that period.

$$u(\Phi) = \sum_{\phi \in \Phi} \sum_{\forall ct \in CT_\Phi} taq_\phi(ct) \tag{9}$$

[1] Using the standard definition of variability of a discrete set X, $\sigma^2(X) = \frac{\sum(x_i - \bar{x})^2}{|X|-1}$

3.7. The multi-objective optimisation problem

The goal of a system of agents G is to maximise $u(\Phi)$ over the lifetime of the WSN. In doing the system should implicitly optimise over the multiple objectives of,

1. Maximising $ea_{\phi}(G)$, the energy available to ensure functionality and coverage of nodes.
2. Maximising $rev_{\phi}(G)$, the distribution of energy to prolong system lifetime.
3. Increase $q_{g,\phi}(at)$, the quality of the individual measurements, at.
4. Maintain coverage $cov_{\Phi}(CT, CT^{*})$ by adapting to node outages and permanent loss.

4. Solving the multi-objective WSN problem

As defined in the previous section, we are looking for a method to optimise multiple objectives in our WSN system. To do so we will incorporate two algorithms, the agent task allocation with risk-impact awareness (ATA-RIA) and multi-group resource allocation optimisation (MG-RAO) algorithms. We give the high-level purpose and requirements of each algorithm in the next sections, however, full details and theoretical justification can be found in Creech et al. (2021a,b).

4.1. Optimisation algorithms for task allocation and resource allocation

The ATA-RIA algorithm enables agents in the system to learn the best actions to take given their current state. This ranges from deciding which other agents to allocate tasks to and obtain the best composite task values, to exploring the system for other agents, while adapting connectivity to handle network disruption. An agent uses the ATA-RIA algorithm to choose an action to take, which can be one of the following,

1. $\texttt{exec}(at)$, The agent will execute the atomic task at itself.
2. $\texttt{alloc}(at, g)$, the agent will allocate the atomic task at to another agent g.
3. $\texttt{info}(g)$, the agent will request information from another agent g.
4. $\texttt{link}(g)$, the agent will allocate resources to hold information on the agent g and maintain a connection.

The ATA-RIA algorithm learns to select the actions that generate the best composite task values, and adapt the choice of action depending on how good the composite task values are in comparison to the historical values through the updating of Q-values in its temporal-update algorithm. The algorithm will select one of the possible actions for the agent g, given it has non-completed, allocated tasks AT, and knows of other agents G through the function,

$$\texttt{ataria}_g : AT \times G \rightarrow \texttt{exec}(AT) \times \texttt{alloc}(AT, G) \times \texttt{info}(G) \times \texttt{link}(G) \tag{10}$$

The MG-RAO algorithm helps agents executing atomic tasks allocate their resources to optimise the corresponding composite task value. This has two parts, an update algorithm that adjust weights of resources based on received atomic task values, $\texttt{mgrao-update}_g : AT \times \mathbb{R} \rightarrow \mathbb{R}$, and the application of these weights to generate the task execution quality itself, $\texttt{mgrao-weight}_g : AT \times \mathbb{R} \rightarrow \mathbb{R}$. The update algorithm will change the resource weightings for an agent, $ar(g, r)$, given the type of an atomic task completed, and the its absolute task value to the corresponding composite task. The weighting algorithm simply returns the resource weighting for calculation of an atomic tasks' quality $q_{g,\phi}(at)$ on its completion.

To enable agents to form arcs we allow atomic tasks that have been allocated to an agent to be either executed by that agent, or re-allocated to further agents, through running the ATA-RIA algorithm . Figure 3 illustrates an arc where there are two re-allocations are made before a specific atomic task is allocated to an agent that completes the task by taking a measurement.

4.2. Algorithm for optimising hierarchical task allocation in networks of agents

The agent networks with hierarchical task allocation optimisation (AN-HTAO) algorithm is defined in Algorithms 1 and 2, which are split for clarity. The flowchart in Figure 4 shows how ~~this~~ agents utilise ATA-RIA and MG-RAO with these recursive actions enabled. We formally define the AN-HTAO algorithm in two parts, Algorithm 1 for sink agents receiving composite tasks, and Algorithm 2 for other agents that form the arc for a task.

In the AN-HTAO (Sink agents) algorithm, Algorithm 1, the sink agent receives a composite task ct comprising of multiple atomic tasks at to be completed. For each of these atomic tasks the sink agent runs the ATA-RIA algorithm to select an action (Line 3). It can execute the atomic task itself (Line 5), or allocate it to another agent it knows about

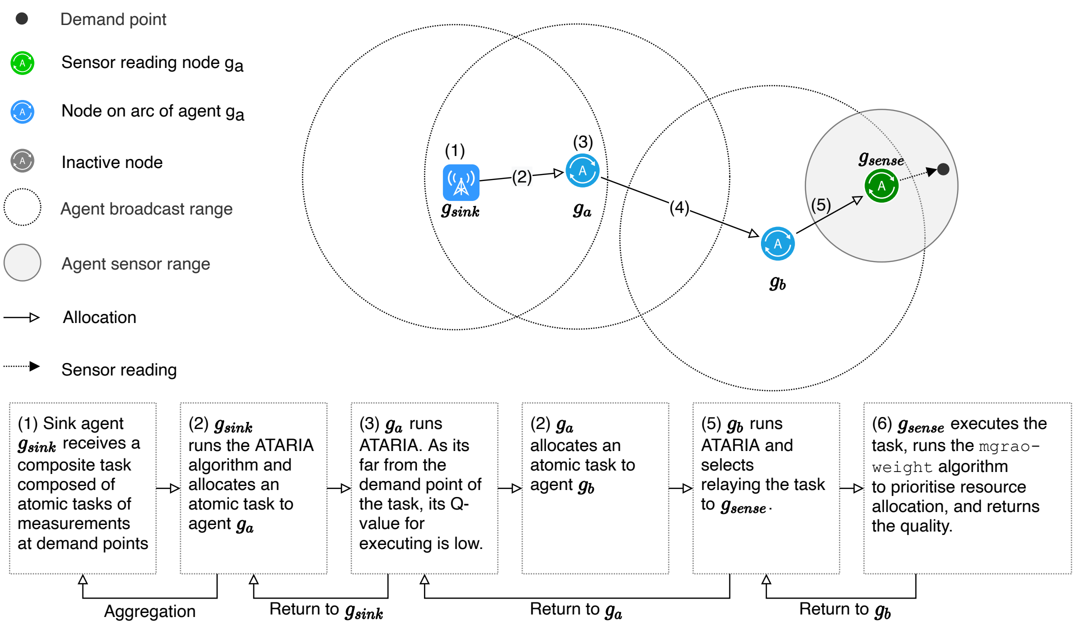
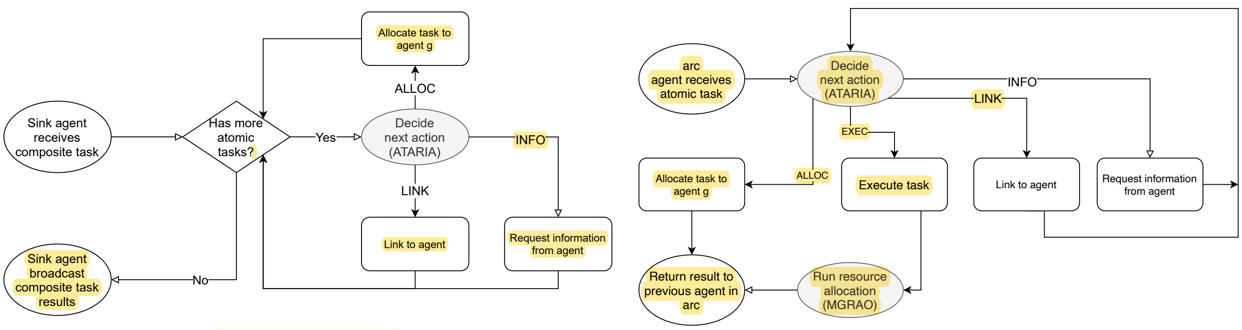


Figure 3: Allocation along an arc. This diagram illustrates how allocations can be relayed along an arc using successive applications of the ATA-RIA algorithm.



(a) Sink agent flow

(b) Arc agent flow

Figure 4: AN-HTAO - Flow chart of combined ATA-RIA/MG-RAO execution. The two algorithms are combined together to allow recursive allocation of tasks and learning of the network.

to complete using the AN-HTAO (Arc agents) algorithm (Line 8) . In both cases, the sink agent receives an atomic task quality value of $q_{g,\phi}(at)$ and removes the atomic task from the list of tasks to complete. If the action selected is one of $\texttt{info}(g)$ or $\texttt{link}(g)$, those actions are carried out without any task executions. Once all atomic tasks have completed, the algorithm can then calculate each atomic tasks' proportional value to the composite task and run the MG-RAO-update algorithm on the last agent in the arc of that atomic task (Lines 16 and 17).

The AN-HTAO (Arc agents) algorithm (Algorithm 2) will complete an atomic task and return its quality, either by executing the task itself (Line 5), or re-allocating to another agent (Line 9). The ATA-RIA algorithm is run and an action selected (Line 3) as in the AN-HTAO (Sink agents) algorithm. This will be repeatedly run until the task in completed an atomic task quality can be returned (Line 5).

5. Evaluation of Environmental Wireless Sensor Networks (E-WSN)

The simulation framework to evaluate the algorithms' performance was based on a realistic deployment scenario as covered by Gomez et al. (2015) and others (Jha et al., 2016; Avram et al., 2017). In this scenario a UAV is used

Algorithm 1: The AN-HTAO (Sink agents) algorithm

Input: ct , The composite task set
Input: g_{self} , The sink agent completing the composite task
Result: $taq_{\phi}(ct)$, The composite task quality of ct

```
// Copy set of atomic tasks to list of incomplete tasks
ctactive ← ct
for at ∈ ctactive do
    // Select action through ATA-RIA
    a ← ataria_g(at, g_self)
    if a = exec(at)? then
        // Execute task at and get an atomic task quality
        q_g,φ(at) ← exec(at)
        // Remove atomic task from incomplete task list
        ctactive ← ctactive − {at}
    else if a = alloc(at, g)? then
        // allocate task at to agent
        q_g,φ(at) ← anhtao-arc(at, g)
        // Remove atomic task from incomplete task list
        ctactive ← ctactive − {at}
    else if a = info(g)? then
        // request information on system agents from agent g
        info(g)
    else if a = link(g)? then
        // allocate resources to information on to agent g and maintaining network connection
        link(g)
end
for at ∈ ct do
    // Target update at the last agent in the arc, the agent that completed the atomic task
    g ← sensor(at)
    // Run MGROA update for target agent using each atomic tasks value
    mgrao-update_g(at, atv(ct, at))
end
return taq_φ(ct)
```

to deploy a large number of sensors over a expansive and remote geographical area, giving an ad-hoc, randomised placement of devices. Solar power cells are used to maintain enough energy to power the sensors over a number of years, given low enough power consumption. We simulated two systems to evaluate the algorithms in four different configurations (See Figure 5).

The simple system equal weighting for each of the CTV components α, β, γ (Eq. 6). The sink node was given 5 measurement tasks to complete from outside the system, repeated 3 times before one episode was complete. 10 nodes were distributed randomly in the system, each capable of completing a task, or allocating it to any of 3 nodes it was connected to. Each node could complete any measurement task with a quality dependent on their closeness to the demand point associated with the task (Eq. 5). The energy of all nodes in the system was fully reset at the end of each episode. An example of the simple system layout can be seen in Figure 5(a).

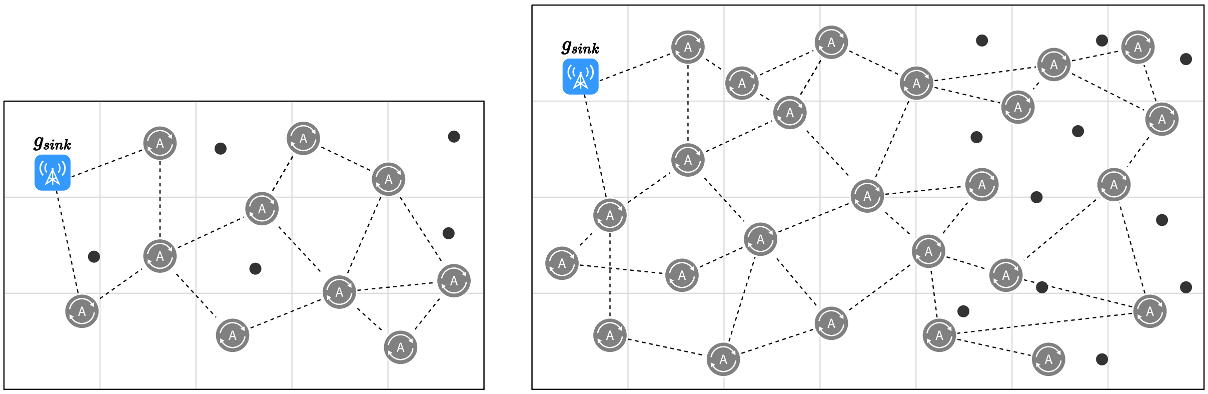
The extended system had CTV component weightings where each of the relevant properties were given an 80% dominance over the value of CTV value. The sink node was given 10 measurements to allocate, with no repetition. It was also placed at a significantly large distance from the demand points associated with the tasks. 25 nodes were distributed randomly in the system. This system examined the impact of the algorithm optimising the allocation of tasks towards the goals stated in Section 3.7. Atomic task quality could be maximised, but at the cost of longer arcs and therefore energy usage, or energy consumption could be minimised, with correspondingly lower task qualities. Figure 6 illustrates these two route types for task completion.

Labels, descriptions, and configurations for each algorithm are shown in Table 1. Results for the balanced algorithm in the simple system, and the energy, quality, distribution algorithms in the extended system are shown in Table 2. System utility percentages show the summed values of composite tasks per episode, as shown in Equation 9, compared

Algorithm 2: The AN-HTAO (Arc agents) algorithm

Input: at , The atomic task to be completed
Input: g_{self} , The agent completing the composite task
Result: $q_{g,\phi}(at)$, The atomic task quality of at

```
taskComplete ← False
while ¬taskComplete do
    // Select action through ATA-RIA
    a ← ataria_g(at, g_self)
    if a = exec(at)? then
        // Execute task at and get an atomic task quality
        q_g,φ(at) ← exec(at)
        taskComplete ← True
    else if a = alloc(at, g)? then
        // allocate task at to agent
        q_g,φ(at) ← anhtao-arc(at, g)
        // allocate task at to agent g
        q_g,φ(at) ← alloc(at, g)
        taskComplete ← True
    else if a = info(g)? then
        // request information on system agents from agent g
        info(g)
    else if a = link(g)? then
        // allocate resources to information on to agent g and maintaining network connection
        link(g)
end
return q_g,φ(at)
```

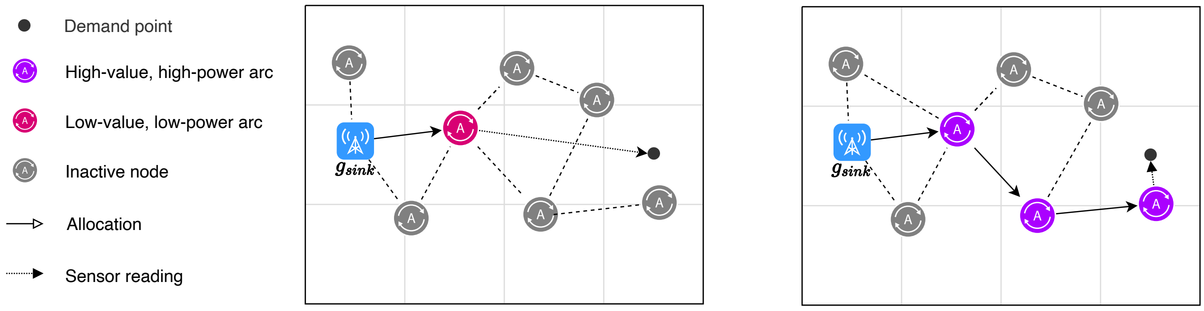


(a) simple system **(b) extended system**

Figure 5: System types. The diagram shows examples of the two systems. In the first simple, there are 10 possible agents that can execute the measurement task, tasks' demand points are distributed across the map. In the second, extended system, there are 25 agents that can execute the measurement tasks. The tasks' demand points are clustered away from the sink node.

to the theoretical maximum utility in the system [2], with the percentage optimisations from the first episode to last in Figure 8. Energy available is presented as a percentage of that of a system containing nodes with full battery charge in Figure 9. We compare the different biases for optimisation across the CTV components using quality-energy balance, task distribution, and arc-depth results. The quality-energy data shown in Figure 10 uses the energy algorithm as a baseline, with the percentage increase or decrease in the average task quality over energy availability components of the CTV equation in Equation 6. The task-distribution in Figure 11 shows the variation in the agents that are completing the tasks, i.e. $|set(G)|/|G|$, with higher values representing more tasks being completed by distinct agents, and lower

[2]Note that the theoretical maximum is not necessarily attainable in all systems, dependent on their randomised node configurations.



(a) Low-value low-power arc **(b) High-value high-power arc**

Figure 6: Routes, power consumption, and task value. The diagram shows two possible arcs possible for the same system. In the first, energy is conserved by having a short arc but the task is completed to a lesser quality. In the second, the maximum quality for the task is achieved, however, there is more energy consumption overall.

Algorithm	**Summary**	**Node count**	**Atomic tasks**	(α, β, γ)
balanced	EWSN optimisation algorithm with balanced objectives.	10	$(5, 5, 5)$	$(0.33, 0.33, 0.33)$
balanced-ext	EWSN optimisation algorithm with balanced objectives.	25	10	$(0.33, 0.33, 0.33)$
energy	EWSN with 80% bias for energy consumption minimisation	25	10	$(0.80, 0.10, 0.10)$
quality	EWSN with 80% bias for task quality maximisation.	25	10	$(0.10, 0.80, 0.10)$
distribution	EWSN with 80% bias for energy distribution maximisation.	25	10	$(0.10, 0.10, 0.80)$

Table 1
Summary of configurations

Algorithm	**CTV % of optimal**	**% of energy available**	**Quality/energy available fraction**	**Average arc depth**	**Energy distribution**
balanced	72%	73%			
balanced-ext	52%	97%			
energy			n/a	2.50	0.488
quality			11%	3.38	0.552
distribution			3%	3.06	0.572

Table 2
Experimental results for each system after 500 episodes

values meaning more agents are completing multiple tasks. Arc depth data in Figure 12 captures how many agents re-allocated each task before it was completed.

As seen in Figure 8, the balanced algorithm optimises the utility by 22% in the simple system and by 14% in the extended system over the 500 episodes. The energy available in the systems overall is shown in Figure 9. As the algorithm improves the routing for task allocation, the energy consumption of the system is reduced, the available energy going from 67% to 73% in the simple system, and by 68% to 97% in the extended one. Note that the task values theoretical maximum becomes less statistically likely to be achievable the larger the system due to the ad-hoc, random layout of the nodes. Importantly however, the task-values themselves are shown to optimised by the algorithm. In contrast, for energy availability, in larger systems there is more possible arc configurations that can complete tasks, and more nodes that are not involved, so the energy consumption as a fraction of energy overall is reduced. In both the simple and extended systems though, we see the energy availability in the system overall increasing with system time.

We now look at the extended system in detail to examine how the algorithm vary the balance of optimisation over the CTV components, allowing multiple-objectives to be targeted. Figure 10 shows the task quality to energy availability ratio. As quality is preferentially optimised for over energy availability, values range higher. The quality algorithm, with its higher γ value, optimises for task values by 13% over the energy algorithm, and the distribution algorithm by 3%. The task distribution of the distribution algorithm, shown in Figure 11, remains relatively steady

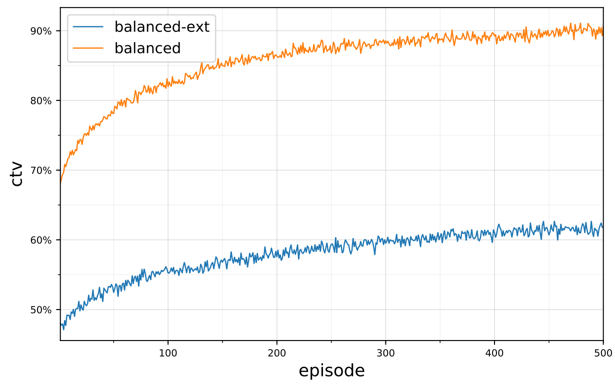


Figure 7: System utility compared to the theoretical maximum for the simple system.

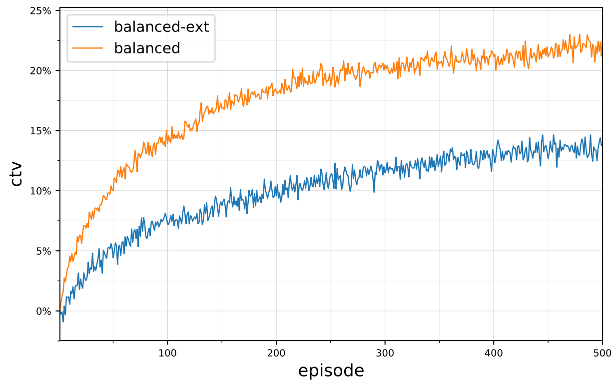


Figure 8: System utility percentage increase over each systems' lifetime.

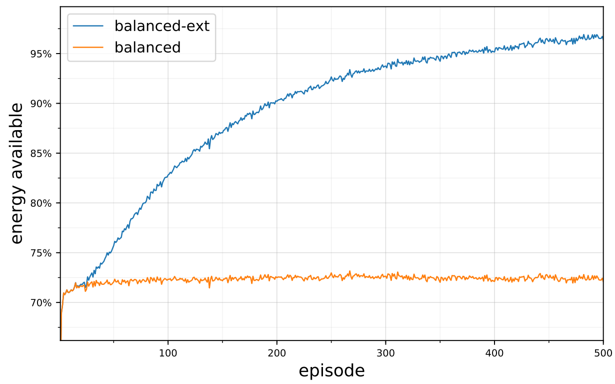


Figure 9: Energy available in the simple and extended systems as percentage of the maximum possible.

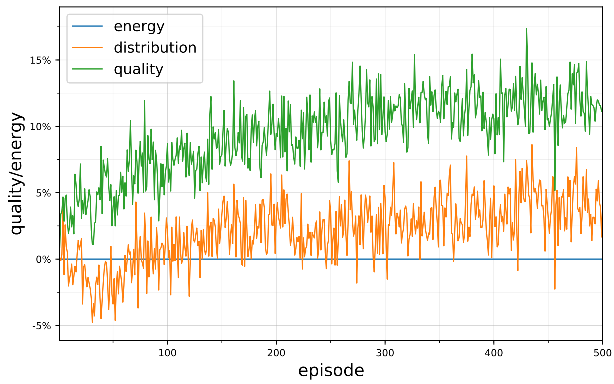


Figure 10: Task quality to energy available ratio in the extended system, with energy as the baseline

throughout the system lifetime at 0.579 to 0.572, dropping only −0.9%. The quality algorithm starts slightly lower than this at 0.560 and falls 1.4% to 0.552. Notably the energy algorithm starts and remains at a significantly lower distribution than both these algorithms at 0.488, dropping 3.4%. Figure 12 shows the related effect on arc-depths for each algorithm. The quality and distribution algorithms have relatively stable arc depths at 3.38 and 3.06 respectively. The average arc-depth of the energy algorithm however drops from 3.02 to 2.50 over the systems' lifetime, a 17.2% decrease.

Relating these results to the systems in Figure 13, the energy algorithm is close to configuration in Figure 13(a), where energy consumption is reduced by using shorter arc-lengths and so, the amount of energy used by agents in the system to broadcast task re-allocations. The cost of this is that the tasks are completed to a lower quality. In Figure 13(c), arc-lengths are large so that tasks can be completed by agents close to their demand points, increasing task quality as well as consumption, as seen in the results for the quality algorithm. The configuration in Figure 13(b) explains the results seen with the distribution algorithm, where tasks are completed by an increased number of different agents with different arc-lengths and task qualities, giving a greater distribution of task completions and energy usage, but with more energy consumption overall than the energy algorithm, and less task-quality than the quality algorithm.

As seen both the simple and extended system configurations, utility and energy availability are both optimised by the algorithm, showing it achieves two of the main goals for applications in WSN. Additionally, the results of the energy, quality, and distribution-optimised algorithms in the extended system show that we can adapt the balance of the CTV components through varying values of α, β, and γ, to achieve an adaptive multi-objective optimisation of the

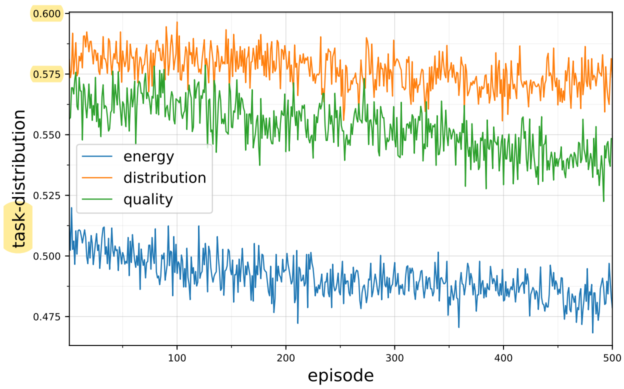


Figure 11: Task distribution in the extended system

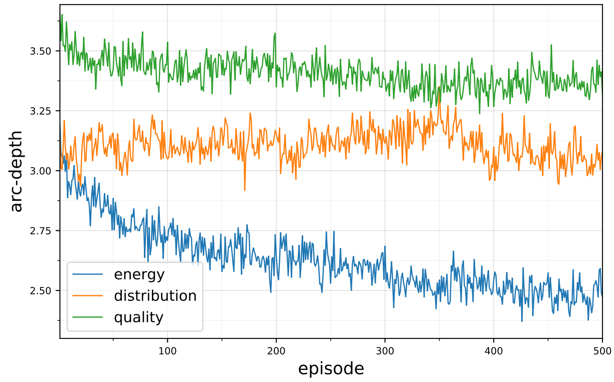
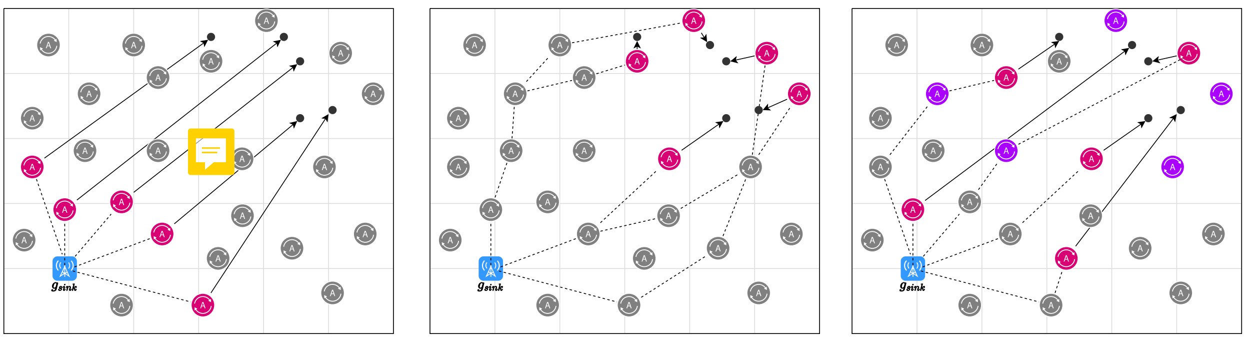


Figure 12: Comparison of arc-depths in the extended system



(a) Energy-optimised system

(b) Quality-optimised system

(c) Distribution-optimised system

Figure 13: Three sample arc patterns in the extended system. In the energy-optimised configuration (a), the sensing agents are near the sync agent. Energy use is minimised but the measurements task-values are low. In the quality-optimised configuration (b) the sensing agents are close to the demand points, maximising the task-values, however there is an increase in energy usage as they are more distant from the sink, with increased arc-length. In the distribution-optimised configuration (c), sensing agents are a mix of close and further away from the demand points, with an the agents participating in the arcs changing with different measurements

system.

Reference to other work algorithms goes here

6. Conclusions and future work

This work detailed and evaluated the novel AN-HTAO algorithm, a combination of the previously described ATA-RIA and MG-RAO algorithms, with extension to hierarchical multi-agent systems. In particular, in their application to wireless sensor network optimisation in dynamic and challenging environments. The algorithm targeted optimisation of the energy available in the system, increasing system lifetime through through task and energy distribution, and the quality of task completion. The algorithm was evaluated on a model WSN system based on a realistic situation where nodes would be randomly dispersed into a geographical area, where maintenance and management is challenging due to harsh or dangerous conditions. The utility of the system was then optimised for sensors taking multiple measurements within the environment, where the quality of the measurement is dependent on the sensors distance to the demand point of the measurement task. Our evaluation showed that the AN-HTAO algorithm optimised the task quality, energy available, and distribution in the system as describe in Section 5, and that these components could be varied in their priorities through altering the α, β and γ values of the CTV function, Equation 6. This allowed the algorithm to balance across these different properties in the given systems and optimise for these multiple objectives in different ratios.

Although the evaluation of the algorithm has been shown in a realistic scenario, the systems represented are still relatively simplistic, and small scale. Future work would look at implementing the algorithm in a larger scale network through simulation, as well as in the real-world, testing how the algorithm performs in a complex environment. There are a number of applications in WSN in which the agents involved are mobile. As both the ATA-RIA and MG-RAO algorithms are designed to handle dynamic environments, where optimisation targets are non-stationary, we expect that the AN-HTAO algorithm will also be able to applicable in these systems. As such evaluation could be extend to simulations with mobile nodes, and tested in real-world vehicle-to-vehicle communications (V2X) systems (Gupta et al., 2017; Tong et al., 2019). We also expect that testing this work in the case of oceanographic monitoring would be a productive next step (Albaladejo et al., 2010b). The combination of harsh environmental conditions, difficulty of providing maintenance for remote sensor nodes, and mobility at slow speeds, should provide ideal conditions for successful practical use of AN-HTAO.

CRediT authorship contribution statement

Niall Creech: Conceptualization of this study, Methodology, Software. **Natalia Craido Pacheco:** Supervision, Writing - Review and Editing. **Simon Miles:** Supervision, Writing - Review and Editing.

References

Ahmed, M.R., Huang, X., Sharma, D., Cui, H., 2012. Wireless Sensor Network: Characteristics and Architectures. URL: https://doi.org/10.5281/zenodo.1072589, doi:10.5281/zenodo.1072589.

Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E., 2002. Wireless sensor networks: A survey. Computer Networks 38, 393–422. doi:10.1016/S1389-1286(01)00302-4.

Albaladejo, C., Sánchez, P., Iborra, A., Soto, F., López, J.A., Torres, R., 2010a. Wireless sensor networks for oceanographic monitoring: A systematic review. Sensors 10, 6948–6968. URL: www.mdpi.com/journal/sensors, doi:10.3390/s100706948.

Albaladejo, C., Sánchez, P., Iborra, A., Soto, F., López, J.A., Torres, R., 2010b. Wireless sensor networks for oceanographic monitoring: A systematic review. Sensors 10, 6948–6968. URL: www.mdpi.com/journal/sensors, doi:10.3390/s100706948.

Anastasi, G., Conti, M., Di Francesco, M., Passarella, A., 2009. Energy conservation in wireless sensor networks: A survey. Ad Hoc Networks 7, 537–568. doi:10.1016/j.adhoc.2008.06.003.

Avram, C., Folea, S., Radu, D., Astilean, A., 2017. Wireless radiation monitoring system. Proceedings - 31st European Conference on Modelling and Simulation, ECMS 2017 , 416–422URL: http://www.scs-europe.net/dlib/2017/ecms2017acceptedpapers/0416-mct{_}ECMS2017{_}0084.pdf.

Babayo, A.A., Anisi, M.H., Ali, I., 2017. A review on energy management schemes in energy harvesting wireless sensor networks. Renewable and Sustainable Energy Reviews 76, 1176–1184. URL: https://www.sciencedirect.com/science/article/pii/S1364032117304598, doi:https://doi.org/10.1016/j.rser.2017.03.124.

Bhandari, S., Bergmann, N., Jurdak, R., Kusy, B., 2017. Time series data analysis of wireless sensor network measurements of temperature. Sensors 17. URL: https://www.mdpi.com/1424-8220/17/6/1221, doi:10.3390/s17061221.

Carlos-Mancilla, M., López-Mellado, E., Siller, M., 2016a. Wireless sensor networks formation: Approaches and techniques. Journal of Sensors 2016. doi:10.1155/2016/2081902.

Carlos-Mancilla, M., López-Mellado, E., Siller, M., 2016b. Wireless Sensor Networks Formation: Approaches and Techniques. Journal of Sensors 2016, 2081902. URL: https://doi.org/10.1155/2016/2081902, doi:10.1155/2016/2081902.

Castillo-effen, M., Quintela, D.H., Jordan, R., Westhoff, W., Moreno, W., 2004. Wireless Sensor Networks for Flash-Flood Alerting , 142–146.

Corke, P., Wark, T., Jurdak, R., Hu, W., Valencia, P., Moore, D., 2010. Environmental wireless sensor networks. Proceedings of the IEEE 98, 1903–1917. doi:10.1109/JPROC.2010.2068530.

Creech, N., Pacheco, N.C., Miles, S., 2021a. Dynamic neighbourhood optimisation for task allocation using multi-agent. arXiv:2102.08307.

Creech, N., Pacheco, N.C., Miles, S., 2021b. Resource allocation in dynamic multiagent systems. arXiv:2102.08317.

Dias, G.M., Nurchis, M., Bellalta, B., 2016. Adapting sampling interval of sensor networks using on-line reinforcement learning, in: 2016 IEEE 3rd World Forum on Internet of Things (WF-IoT), pp. 460–465. doi:10.1109/WF-IoT.2016.7845391.

Engmann, F., Katsriku, F.A., Abdulai, J.D., Adu-Manu, K.S., Banaseka, F.K., 2018. Prolonging the Lifetime of Wireless Sensor Networks: A Review of Current Techniques. Wireless Communications and Mobile Computing 2018. doi:10.1155/2018/8035065.

Escolar, S., Chessa, S., Carretero, J., 2014. Energy management in solar cells powered wireless sensor networks for quality of service optimization. Personal and Ubiquitous Computing 18, 449–464. URL: https://doi.org/10.1007/s00779-013-0663-1, doi:10.1007/s00779-013-0663-1.

Fang, P., Zhou, Y.Z., Xin, Z., 2010. A Remote Wireless Sensor Networks for Water Quality Monitoring. CICC-ITOE 2010 - 2010 International Conference on Innovative Computing and Communication, 2010 Asia-Pacific Conference on Information Technology and Ocean Engineering , 364–365doi:10.1109/CICC-ITOE.2010.9.

Goel, K., Bindal, A.K., 2018. Wireless sensor network in precision agriculture: A survey report, in: 2018 Fifth International Conference on Parallel, Distributed and Grid Computing (PDGC), pp. 176–181. doi:10.1109/PDGC.2018.8745854.

Gomez, A., Lagadec, M.F., Magno, M., Benini, L., 2015. Self-powered wireless sensor nodes for monitoring radioactivity in contaminated areas using unmanned aerial vehicles, in: SAS 2015 - 2015 IEEE Sensors Applications Symposium, Proceedings, Institute of Electrical and Electronics Engineers Inc. doi:10.1109/SAS.2015.7133627.

Gupta, A., Singh, R., Ather, D., Shukla, R.S., 2017. Comparison of various routing algorithms for VANETS. Proceedings of the 5th International Conference on System Modeling and Advancement in Research Trends, SMART 2016 , 153–157doi:10.1109/SYSMART.2016.7894509.

Gupta, S.K., Sinha, P., 2014. Overview of Wireless Sensor Network: A Survey. International Journal of Advanced Research in Computer and Communication Engineering 3.

Iqbal, M., Naeem, M., Anpalagan, A., Ahmed, A., Azam, M., 2015. Wireless sensor network optimization: Multi-objective paradigm. Sensors 15, 17572–17620. URL: https://www.mdpi.com/1424-8220/15/7/17572, doi:10.3390/s150717572.

Jha, D., Dahal, S., Shukla, S., Shahi, B., Student, P.G., 2016. Radioactive Contamination Detection in Water Using Wireless Sensor Network 1, 2395–4396. URL: www.ijariie.com.

Kandris, D., Nakas, C., Vomvas, D., Koulouras, G., 2020. Applications of wireless sensor networks: An up-to-date survey. Applied System Innovation 3, 1–24. doi:10.3390/asi3010014.

Khan, M.I., Rinner, B., 2014. Performance analysis of resource-aware task scheduling methods in wireless sensor networks. International Journal of Distributed Sensor Networks 10, 765182. URL: https://doi.org/10.1155/2014/765182, doi:10.1155/2014/765182, arXiv:https://doi.org/10.1155/2014/765182.

Kulkarni, R.V., Förster, A., Venayagamoorthy, G.K., 2011. Computational intelligence in wireless sensor networks: A survey. IEEE Communications Surveys Tutorials 13, 68–96. doi:10.1109/SURV.2011.040310.00002.

Liu, M., Cao, J., Chen, G., Wang, X., 2009. An energy-aware routing protocol in wireless sensor networks. Sensors 9, 445–462. URL: https://www.mdpi.com/1424-8220/9/1/445, doi:10.3390/s90100445.

Mahdy, A.M., 2008. Marine wireless sensor networks: Challenges and applications. Proceedings - 7th International Conference on Networking, ICN 2008 , 530–535doi:10.1109/ICN.2008.115.

Mak, N.H., Seah, W.K., 2009. How long is the lifetime of a wireless sensor network? Proceedings - International Conference on Advanced Information Networking and Applications, AINA , 763–770doi:10.1109/AINA.2009.138.

Mihaylov, M., Borgne, Y.A.L., Tuyls, K., Nowé, A., 2012. Decentralised reinforcement learning for energy-efficient scheduling in wireless sensor networks. Int. J. Commun. Netw. Distrib. Syst. 9, 207–224. URL: https://doi.org/10.1504/IJCNDS.2012.048871, doi:10.1504/IJCNDS.2012.048871.

Mihaylov, M., Tuyls, K., Nowé, A., 2010. Decentralized learning in wireless sensor networks, in: Taylor, M.E., Tuyls, K. (Eds.), Adaptive and Learning Agents, Springer Berlin Heidelberg, Berlin, Heidelberg. pp. 60–73.

Min, R., Bhardwaj, M., Seong-Hwan Cho, Shih, E., Sinha, A., Wang, A., Chandrakasan, A., 2001. Low-power wireless sensor networks, in: VLSI Design 2001. Fourteenth International Conference on VLSI Design, pp. 205–210. doi:10.1109/ICVD.2001.902661.

Munir, S.A., Ren, B., Jiao, W., Wang, B., Xie, D., Ma, J., 2007. Mobile wireless sensor network: Architecture and enabling technologies for ubiquitous computing, in: 21st International Conference on Advanced Information Networking and Applications Workshops (AINAW'07), pp. 113–120. doi:10.1109/AINAW.2007.257.

Nurmi, P., . Reinforcement Learning for Routing in Ad Hoc Networks .

Oliveira, L.M., Rodrigues, J.J., 2011. Wireless sensor networks: A survey on environmental monitoring. Journal of Communications 6, 143–151. URL: http://www.jocm.us/index.php?m=content{&}c=index{&}a=show{&}catid=54{&}id=202, doi:10.4304/jcm.6.2.143-151.

Paradis, L., Han, Q., 2007. A survey of fault management in wireless sensor networks. Journal of Network and Systems Management 15, 171–190. doi:10.1007/s10922-007-9062-0.

Prauzek, M., Konecny, J., Borova, M., Janosova, K., Hlavica, J., Musilek, P., 2018. Energy harvesting sources, storage devices and system topologies for environmental wireless sensor networks: A review. Sensors (Switzerland) 18. doi:10.3390/s18082446.

Praveen Kumar, D., Amgoth, T., Annavarapu, C.S.R., 2019. Machine learning algorithms for wireless sensor networks: A survey. Information Fusion 49, 1–25. URL: https://linkinghub.elsevier.com/retrieve/pii/S156625351830277X, doi:10.1016/J.INFFUS.2018.09.013.

Predd, J.B., Kulkarni, S.R., Poor, H.V., 2006. Distributed learning in wireless sensor networks. doi:10.1109/MSP.2006.1657817, arXiv:0503072.

Qian, C., Qi, H., 2008. Coverage estimation in the presence of occlusions for visual sensor networks, in: Nikoletseas, S.E., Chlebus, B.S., Johnson, D.B., Krishnamachari, B. (Eds.), Distributed Computing in Sensor Systems, Springer Berlin Heidelberg, Berlin, Heidelberg. pp. 346–356.

Ramasamy, V., 2017. Mobile wireless sensor networks: An overview. Wireless Sensor Networks—Insights and Innovations .

Sengupta, S., Das, S., Nasir, M., Panigrahi, B., 2013. Multi-objective node deployment in wsns: In search of an optimal trade-off among coverage, lifetime, energy consumption, and connectivity. Engineering Applications of Artificial Intelligence 26, 405–416. URL: https://www.sciencedirect.com/science/article/pii/S0952197612001248, doi:https://doi.org/10.1016/j.engappai.2012.05.018.

Sharma, A., Chauhan, S., 2020. A distributed reinforcement learning based sensor node scheduling algorithm for coverage and connectivity maintenance in wireless sensor network. Wireless Networks 26, 4411–4429. URL: https://doi.org/10.1007/s11276-020-02350-y, doi:10.1007/s11276-020-02350-y.

Sonavane, S.S., Kumar, V., Patil, B.P., 2008. Designing wireless sensor network with low cost and low power, in: 2008 16th IEEE International Conference on Networks, pp. 1–5. doi:10.1109/ICON.2008.4772585.

Tong, W., Hussain, A., Bo, W.X., Maharjan, S., 2019. Artificial Intelligence for Vehicle-To-Everything: A Survey. IEEE Access 7, 10823–10843. doi:10.1109/ACCESS.2019.2891073.

Warneke, B.A., Pister, K.S.J., 2002. Mems for distributed wireless sensor networks, in: 9th International Conference on Electronics, Circuits and Systems, pp. 291–294 vol.1. doi:10.1109/ICECS.2002.1045391.

Werner-Allen, G., Lorincz, K., Welsh, M., Marcillo, O., Johnson, J., Ruiz, M., Lees, J., 2006. Deploying a wireless sensor network on an active volcano. IEEE Internet Computing 10, 18–25. doi:10.1109/MIC.2006.26.

Wu, F., Tan, C.W., Sarvi, M., Rudiger, C., Yuce, M.R., 2017. Design and implementation of a low-power wireless sensor network platform based on xbee, in: 2017 IEEE 85th Vehicular Technology Conference (VTC Spring), pp. 1–5. doi:10.1109/VTCSpring.2017.8108667.

Xu, G., Shen, W., Wang, X., 2014. Marine environment monitoring using wireless sensor networks: A systematic review, in: 2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 13–18. doi:10.1109/SMC.2014.6973877.

Yick, J., Mukherjee, B., Ghosal, D., 2008. Wireless sensor network survey. Computer Networks doi:10.1016/j.comnet.2008.04.002.

Đurišić, M.P., Tafa, Z., Dimić, G., Milutinović, V., 2012. A survey of military applications of wireless sensor networks, in: 2012 Mediterranean Conference on Embedded Computing (MECO), pp. 196–199.